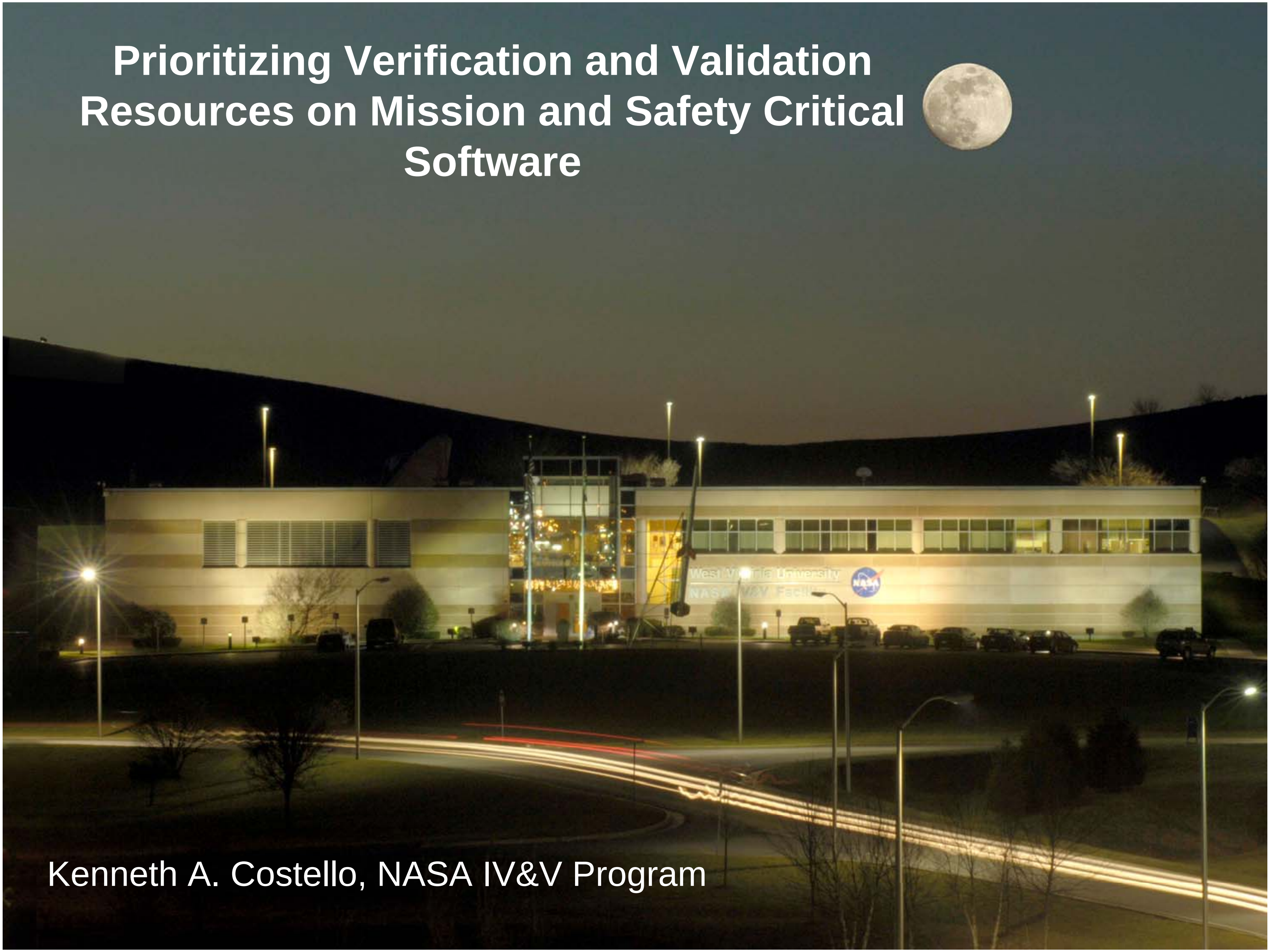
Prioritizing Verification and Validation Resources on Mission and Safety Critical Software



Kenneth A. Costello, NASA IV&V Program

Discussion Outline

- Quick Overview of the NASA Approach to IV&V
  - What is IV&V?
  - Objectives
  - Goal
- NASA Approach to Determining IV&V Tasking
  - Software Integrity Level Assessment Process
  - Developing the task list
- Dealing with IV&V Implementation Constraints
  - Managing Risk with SILAP
  - Slicing the Data
  - Data views
- Future

What is IV&V at NASA?

- An engineering discipline employing rigorous methods for evaluating the correctness and quality of the software product throughout the software life cycle from a system level viewpoint
- The NASA Software IV&V approach covers not only expected operating conditions but the full spectrum of the system and its interfaces in the face of unexpected operating conditions or inputs
- Three nominal aspects are technical, managerial and financial
  - Technical – The IV&V Program has its own approach to determining where to focus analysis activities
  - Managerial – The IV&V Program is functionally managed by the NASA Office of Safety and Mission Assurance while project level management comes form the IV&V Program
  - Financial – The IV&V Program is funded from NASA Corporate General & Administrative funds for Agency identified high priority Projects

A Lifecycle Approach

- The NASA approach is not end of life cycle testing
- It is a testing approach
- Each task that is performed “tests” a development artifact ranging from system and software requirements to source code and test results
- Each task in the NASA IV&V Work Breakdown Structure is designed to “test” a development artifact or process
- Testing is throughout the life cycle

Objectives of IV&V

- To find defects within the system with a focus on software and its interactions with the system
- To make an assessment of whether or not the system is usable in an operational environment, again with a focus on the software within the system
- To identify any latent risks associated with the software

Ultimate Goal of IV&V

Establish confidence that the software is fit for its purpose within the context of the system

- Note that the software may not be free from defects
  - That is rarely the case and can be difficult to prove
- However, the software must be sufficient for its intended use
  - The system should be as described by the requirements
  - The requirements should be representative of the user’s needs
- The type of use will determine the level of confidence that is needed
  - The consequence of software defect/failure will drive the needed level of confidence

Approach to Determining IV&V Tasking

Software Integrity Level Assessment Process

- The NASA Software Integrity Level Assessment Process (SILAP)
  - Combination of the best of the current processes and best practices from industry and academia
  - The process has been used on over 15 projects and executed multiple times with excellent results
- Process supports two key goals
  - Provides the NASA IV&V Program with a consistent risk-based approach to determining IV&V tasking
  - An extremely effective communication tool with the development projects

SILAP Overview

- Measure of software goodness
  - Software integrity level
    - How good (dependable, reliable, etc.) does a software component need to be in terms of its role in the system
      - Understand how the component fits within and impacts the system
      - Understand what is required of that component to be able to maintain the functionality of the system
  - Software integrity level is equivalent to the tolerable level of risk that is associated with the system.
    - A software component can be associated with risk because
      - a failure (or defect) can lead to a threat, or
      - its functionality includes mitigation of consequences of initiating events in the system’s environment that can lead to a threat

SILAP Overview (2)

- Determine overall needed integrity level
- Impact of a defect
  - Determine the worse case error (at the software component level) that has a reasonable or credible fault/failure scenario
  - Consider the system architecture and try to understand how that software fault/failure scenario may affect the system
  - This determines the Consequence of the defect
- Probability that the developer may insert an error into a software component is determined
  - Not an assessment of the probability of a failure, but rather the probability that an error of any type may exist in the operational software
  - This determination is known as the Error Potential

SILAP Overview (3)

- These two factors provided a risk-like look at the software components
  - However, there is no relationship defined between Consequence and Error Potential that results in a risk level
  - Rather Consequence and EP are addressed independently of each other as will be shown later
- This risk-based approach is the prime reason that communications with development projects have grown better
- The risk-based approach also allows the process to be used to define different levels of risk reduction

Consequence

- Consequence consists of the following items
  - Human Safety – This is a measure of the impact that a failure of this component would have on human life
  - Asset Safety – This is a measure of the impact that a failure would have on hardware
  - Performance – This is a measure of the impact that a failure would have on a mission being able to meet its goals

Error Potential

- Error Potential consists of the following items
  - Developer Characteristics
    - Experience – This is a measure of the system developer’s experience in developing similar systems
    - Organization – This is a measure of the complexity of the organization developing the system (distance and number of organizations involved tend to increase the probability of errors being introduced into the system)
  - Software/System Characteristics
    - Difficulty – This is a measure of the complexity of the software being developed
    - Degree of Innovation – This is a measure of the level of innovation needed in order to develop this system/software
    - System Size – This is a measurement of the size of the system in terms of the software (i.e., Source Lines of Code)
  - Development Process Characteristics
    - Formality of the Process – This is a measure of how maturity of the developer’s processes
    - Re-use Approach – This is a measure of the level of re-use for the system/software
    - Artifact Maturity – This is a measure of the current state of the development documentation in relation to the state of the overall development project (i.e., the is past critical design review but the requirements documents are still full of TBDs and incompletes)

Developing Tasking

- Once the scoring is complete a tasking set is developed based on each individual score
  - There is a set of tasking associated with a given Consequence score
  - There is a set of tasking associated with a given Error Potential score
- The tasks are then combined into one set of tasks for that component
  - The tasks are not exclusive to a given score, i.e., you may have a task that shows up for the Consequence score and for the Error Potential score
- This results in a matrix of software components and scores that provides the starting set of requirements or tasks for IV&V on that project

SILAP Process

- More specific details of the SILAP process can be found at the NASA IV&V Program IV&V Management System (IMS) website
- http://ims.ivv.nasa.gov
- Look under the documentation link for Work Instruction 9-8-1
    - At the time of the writing of this presentation the work instruction had not been approved for publishing

Dealing with Implementation Constraints

Managing Risk With SILAP

- The SILAP has been used effectively on many different types of projects and across several iterations
  - The process is easy to use and generates data that is understandable on many levels, i.e., technical and managerial
  - The data can also be grouped to provide different views of the risk associated with a system
  - These different views provides differing level of risk reduction but can still be tuned to meet the goals and objectives of the IV&V Program

Managing Risk With SILAP (2)

- As an example, the SILAP data can provide viewpoints that allow for differing levels of financial commitment but still meet the objectives of the IV&V program (although at the most minimum level)
  - The goal in developing the viewpoints was to always keen in mind a desire to meet our primary objective but with a funding constraint
  - Several other options to dealing with a funding constraint were also explored
  - However the approach using SILAP provided the most consistency in application and management

Managing Risk with SILAP (3)

- The approach to developing the viewpoints was to start with one of the underlying principles in SILAP; that the identified software components could be ranked in a priority fashion
  - Items of high consequence and error potential were more important than those of lower values
- The foundation of the process is determining the consequence of a defect in a software component
  - Performing IV&V on items of high consequence is generally more important than performing IV&V on items of higher error potential
  - Also keep in mind that the tasking associated with each factor, consequence and error potential, are disjoint and separate
- So with this in mind, not only can components be ranked by a combination of the two scores, but also by each score individually

Slicing the Data

- Consequence is the foundation
  - It is important to examine software components with high consequence irrespective of their error potential
  - It is less important to examine low consequence components irrespective of their error potential
- Several ranking approaches were developed
- To understand these approaches, there was also a need to understand some of the details in the criteria for scoring consequence
- In the table on the next page, the shaded areas represent scores of strong consequence
  - In other words, serious or greater injury, permanent loss of an important part of the spacecraft, or a permanent loss of primary mission data

Consequence Criteria

	Consequence				
	1	2	3	4	5
Human Safety	No impact to human safety	Discomfort or nuisance	Minor injury or potential for minor injury	Major injury or potential for major injury	Loss of life
Asset Safety	No damage to any s/c component Short-term partial loss of other critical asset	Temporary loss to a s/c component Partial loss of other critical asset Degradation of a non-primary component	Permanent loss of non-primary component Complete loss of other critical asset Degradation of a primary component	Permanent loss of primary component	Complete loss of vehicle/spacecraft/system
Performance	Unrecoverable loss of non-primary mission/science data	Unable to achieve non-primary mission/science objective Loss of non-primary capability	Unable to achieve a particular primary mission/science objective Unrecoverable loss of primary missions/science data	Unable to achieve multiple mission/science objectives Loss of primary capability	Unable to achieve minimum mission/science objectives or minimum success criteria

Slicing the Data (2)

- Understanding this breakdown between strong and weak consequences creates a basis for further refining an IV&V approach
- With the primary objective being to provide confidence that the mission will succeed, we can use the strong consequences as delineators in this definition
  - That is to say, the minimal level of confidence in a mission is defined by saying that it will not cause a major injury or worse, destroy a major spacecraft component or the entire spacecraft, or cause the permanent loss of data or a complete inability to meet one or more mission objectives.
- From this statement several viewpoints were developed based on differing rankings of consequence and error potential

Data Views

- Each view brings a different level of risk reduction but are easily determined simply by sorting the scores for components
  - Consequence >2 along with EP tasking
  - Consequence > 2 with no EP tasking
  - No EP tasking
  - Performance > 2, Asset safety > 3
  - Performance > 2, Asset safety > 3 with no EP tasking
- Note that Performance and Asset safety are called out
  - Creates a viewpoint based on the components of Consequence
  - This was needed due to the weighting scheme used; combinations of scores exist that may place the individual component in the shaded area on the previous chart, but the overall resulting Consequence score may be a 2
  - Also note that for the purpose of this example human safety score were not included
- For this presentation, a generic project was developed and the following potential cost savings were generated based on these scores
- The project had 44 identified software components and the savings are based on the average cost of performing a given task from the WBS

Bands of Potential Funding

	% Savings	#of components out
Full Tasking	-	0
PF > 2, AS > 3	13%	11
EP out	39%	0
PF > 2, AS > 3 and EP out	42%	11
Consequence > 2	40%	29
Consequence > 2 and EP out	50%	29

- The cost reduction is taken against the nominal cost for doing all of the tasking
- The components out column shows the number of components removed form the identified 44

- In each of these cases, our current belief is that we can still meet the desire of NASA in providing confidence in the most important portions of the mission
- It is not the best possible risk reduction, but it meets the minimal needs of NASA

The Future of SILAP

- The SILAP is the planned approach for determining IV&V tasking for the future
- Further investigations are also underway to examine the scoring and document and analyze any trends that can be found
- A tool is currently being developed to allow for the easy capture and analysis of the scoring data for each individual software component
- Other work is also ongoing to determine how to integrate the results of SILAP into other views of the systems using various modeling techniques
  - The goal is to provide even greater focus on the critical components of the system and show how they relate to the success or failure of the system
  - Provide a more detailed approach to performing validation of software using a model based analysis approach

Questions?

- If you have any further questions or comments you can contact

Dr. Butch Caffall, IV&V Program Manager, Director IV&V Facility
NASA IV&V Facility
304 367 8201
Butch.Caffall@nasa.gov

Kenneth Costello, IV&V Program Chief Engineer
NASA IV&V Facility
304 367 8343
Kenneth.A.Costello@nasa.gov

Christina Moats, IV&V Planning and Scoping Lead
NASA IV&V Facility
304 367 8340
Christina.D.Moats@nasa.gov